Detection of Missing Tweets based on Browsing Interval and Topic Granularity

Hiromitu Ohara
Konan University
Okamoto 8–9–1
Higashinada–ku Kobe, Japan
m1424003@center.konan-u.ac.jp

Yu Suzuki
Nara Institute of Science and Technology
891605 Takayama-cho, Ikoma,
Nara, 630-0192, Japan
ysuzuki@is.naist.jp

Akiyo Nadamoto
Konan University
Okamoto 8–9–1
Higashinada–ku Kobe, Japan
nadamoto@konan-u.ac.jp

ABSTRACT

Twitter users who browse tweets can follow other users in whom they are interested. They can obtain interesting information from other users' tweets on their timeline. If they follow many users, then they can expect numerous tweets on their timeline. However, if users do not browse their timeline for some time, they can lose interesting and important information. Therefore, a system that automatically presents a summary of lost information can be extremely beneficial. As described herein, we propose a method of extracting lost information automatically based on a user's browsing time interval and the topic structure of a followee's tweets. First, we classify a followee's tweets that contain the user's missing information, and assign topics to the groups. Next, we generate a topic graph based on the semantic structure from Wikipedia. We decide whether the tweet groups are missed using the followee's topic graph based on the browsing time interval. Finally, we extract missing information and present it to the user.

Categories and Subject Descriptors

H.4.m [**Information Systems**]: Miscellaneous; I.7.m [**Document and Text Processing**]: Miscellaneous

Keywords

Microblog, Twitter, Topic Granularity

1. INTRODUCTION

Microblogs have become extremely popular: users can easily post and obtain various information from the internet. Twitter, an extremely popular microblog, accumulates and distributes vast amounts of information. A Twitter user who browses tweets can follow other users, called followees, in whom the user is interested. The user can then receive interesting information from followees' tweets on the user's own timeline. When a user follows many followees, the user can browse numerous and diverse tweets on the user's own timeline. However, a user who cannot browse for some period of time loses interesting and important information. Uninteresting information and known data are also included in the great numbers of missing tweets. Therefore, large numbers of missing tweets can be expected to include important, unimportant, known topic, and unknown topic information. To obtain the interesting and important information that is lost, but included in the large amount of missing tweets, a user must search for the tweets that were posted while the user was unable to browse. Then a user must read and comprehend the contents of the missing tweets. That is difficult work for users. For such cases, we consider a system that can be beneficial for users: it browses a summary of lost information automatically. We propose a method to extract this lost information, which is important but unknown information. It browses a summary of these numerous tweets automatically. As described in this paper, we designate the tweets which a user has lost and important information as "missing information," the target user for whom we extract missing information as a "browsing user," and the person whom the browsing user follows as a "followee."

As described in this paper, as a first step of presenting missing information of tweets, we propose a method to extract missing information and to present contents based on types of missing information. The user interface used for presenting the missing information is left as a subject for future work.

This paper is organized as follows: Section 2 discusses related work; Section 3 presents definition of browsing interval and topic granularity; Section 4 presents a method to extract missing information based on topic granularity and time interval; Section 5 discusses experiment results. Section 6 presents conclusions of our study.

2. RELATED WORKS

Topic detection on Twitter has been studied by many researchers. Mathioudakis et al.[12] propose a system detecting emerging topics on twitter by detecting bursty keywords in tweets. Becker et al.[1] propose a method for identifying tweets about real world events and detecting trending topics. Lee et al. [9] try to detect trending topics on twitter by using text-based classification and network-based classification. These method attend emerging topics, but our method target not only emerging topics but also normal topics like authors hobby,job and experience in daily life. Hong et al. [6] propose topic modeling in Twitter using

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

iiWAS '15, December 11-13, 2015, Brussels, Belgium

© 2015 ACM. ISBN 978-1-4503-3491-4/15/12…$15.00

DOI: http://dx.doi.org/10.1145/2837185.2837233

LDA(Latent dirichlet allocation[3]) and the Author-Topic model. Michelson et al. [13] try to detect Topics using author information of tweets with categories in Wikipedia. Xu et al.[19] try to extract topics in tweet with article and link histories in Wikipedia. Kasiviswanathan et al. [8] detect topics from Twitter using the dictionary learning method. Bernstein et al.[2] try to extract relevant topics of tweets based on number of search engines result. However, these studies do not examine topic granularity and time information. Cataldi et al.[4] propose a topic detection method with relation of topics by author information of tweets and the topic life cycle. They specifically examine time information for all of Twitter, but we particularly examine time information only for the viewer timeline to facilitate extraction of more specific information.

In our method, we extract topic granularity by clustering tweet and detecting topics for each clusters. Sriam et al. [17] propose classify tweets into categories such as News, Events, Opinions, and so on. They use author information and features, but their method is not address tweets topic. Rosa et al. [15] verify supervised clustering methods and unsupervised clustering methods. Their results show that is difficult to cluster information accurately using unsupervised methods such as standard LDA or K-means clustering[10] because tweets are short and Twitter contain to many noise. We use Repeated-Bisection clustering. It is an unsupervised method, but it is regarded as an effective method for short texts.

In recent years, many studies specifically examine time information related to Twitter. Ren et al.[14] propose a topic model for summarization of personalized useful information related to Twitter based on the association between users and time-aware summarization. Sasaki et al.[16] emphasizes the study of variation in topic trends by time, proposing a topic model improved on Twitter-LDA[20]. Magdy et al.[11] present a searching relevant summarized information from twitter by posted time and events in real world. However, these methods do not address topic granularity.

3. DEFINITION OF BROWSING INTERVAL AND TOPIC GRANULARITY

3.1 Definition of browsing interval

We consider that there are two types of missing information that are partially-known information and complementary-unknown information. We call the former "known topic" the latter "unknown topic." The definition of "known topics" and "unknown topics" as follows.

- Known topic
  A known topic is a topic of tweets. Some tweets are posted during a browsing user browsing time; some tweets are posted during a user non-browsing time. Then, a browsing user knows part of the topic.

- Unknown topic
  An unknown topic is a topic of tweets posted only during the user non-browsing period. A browsing user does not know information at all.

Figure 1 presents an image showing the known topic and unknown topic. In Figure 1, the followee tweets about part of "baseball" and "research" during a browsing user's browsing time and the browsing user know part of "baseball" and "research" topics. They become known topics. On the other hands, the followee tweets about "football" during a browsing user's un-browsing time, the browsing user does not know about information of "football" which is followee tweeted at all. The topic of "football" becomes unknown topic. We consider, a browsing user can understand roughly known topic information that are tweeted his/her un-browsing time, because he/she already browsed the same topic of tweets posted during the user browsing time. In this case, we consider it is good for the browsing user to present outline of the topics, then we present the topic structure of the missing information to him/her. On the other hands, for an unknown topic, it is difficult for a browsing user to understand all contents of topic clearly because the tweets were tweeted during a time when the browsing user cannot browse. We consider that it is necessary to present information that has greater detail that a user can discover the full breadth of the topic than in the case of a known topic.

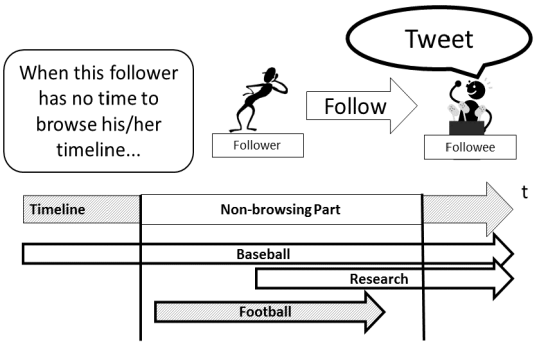

Figure 1: Image of the known topic and unknown topic.

3.2 Definition of topic granularity

It is important to consider topic granularity because missing information that should be presented for browsing users is based on browsed topic details. "Ichiro" and "Masahiro Tanaka" are Japanese major league baseball players. Tweets related to "Ichiro" and tweets related to "Masahiro Tanaka" are classified as "Japanese Major Leaguers," including also "Major Leaguers" such as "Jose Altuve" and "John Donaldson." When classifying these tweets according to greater granularity such as "baseball," the case includes tweets about topics such as "baseball team," "baseball stadium," and so on. However, we can classify these tweets according to finer granularity such as "Japanese major leaguer," but the case includes only "Ichiro" and "Masahiro Tanaka." For example, the browsing user did not browse all the followee's tweets related to "Ichiro" and "Masahiro Tanaka," but the browsing user browses some tweets about "John Donaldson." In this case, when we classify the followee's tweets to "Japanese Major Leaguers," it becomes an unknown topic. When we classify them to "Major Leaguers" or "Baseball," they becomes known topics. Because the topic differs from granularity of the classification (see Fig. 2), we must consider the granularity of the topic. When extracting missing information automatically, it is important to consider topic granularity.

4. EXTRACTING MISSING INFORMATION

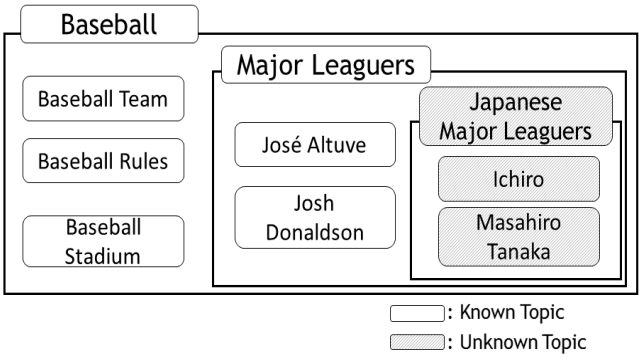

Figure 2: Granularity of topics.

We consider both the user's browsing time and topic granularity to extract missing information. As described herein, we propose a method to extract this missing information automatically based on the browsing time interval and topic granularity. First, we calculate the topic granularity. Next we divide nodes of a topic graph based on the browsing time interval. Then we extract missing information of two types and browse it.

The following and Fig. 3 present details for the flow of extracting missing information.

1. Considering topic granularity
   - 1–1 Extracting tweets that the followee tweeted during the time when the browsing user did not browse and before and after that time
   - 1–2 Clustering tweets and extract topics in each cluster
   - 1–3 Creating the topic graph using a semantic structure in Wikipedia
2. Divide the topics in the topic graph based on the browsing time interval
   - 2–1 Extracting the tweet time for each tweet in the topic graph
   - 2–2 Dividing the topic into known and unknown topics based on the browsing time interval
3. Extracting missing information and dividing it into known topics and unknown topics based on topic graphs
4. Presenting missing information in each type

4.1 Topic granularity

4.1.1 Clustering tweets

We extract tweet topics using clustering tools. When we cluster the tweets, we use feature words of the tweets.

Feature words

To classify and detect the topic from tweets, we use feature words, which are proper nouns, general nouns, and unknown words. The reason for using unknown words is to adapt new words and abbreviated words that are used on Twitter. Candidates of feature words have more general words such

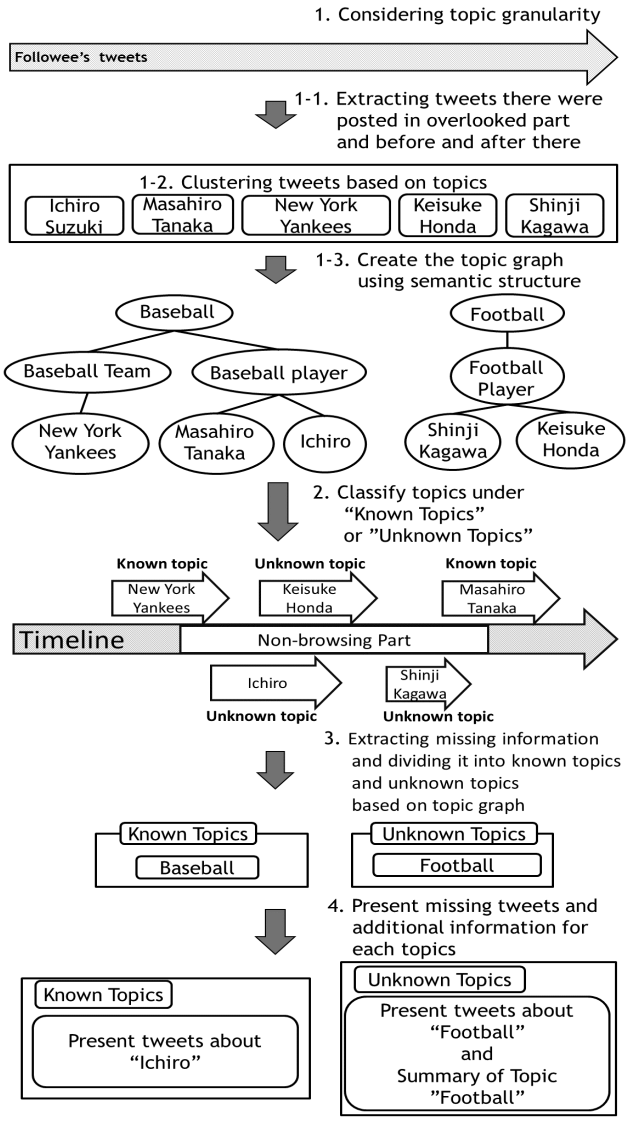

Figure 3: System flow.

as "game" and "player," which are not specific to the topic of kinds of sports. They also have newly coined words that are unique to a community. We delete newly coined words, more general words, and those from candidates of feature words. More general words are used very often on the internet. Newly coined words are not used outside the community. We consider more general words as those which are often used in Wikipedia articles. In contrast, newly coined words are not used in Wikipedia articles. We regard more general words as those used in more than 100 Wikipedia articles. Newly coined words are those which are not used in any Wikipedia article. Consequently, feature words are proper nouns, general nouns, and unknown words used in 1–100 Wikipedia articles. The vector of feature words is the value of IDF when a tweet is regarded as one document.

Clustering

When classifying tweets using feature words to extract tweets topics, we use Repeated Bisection[18], which is a method

used for bayon[1] and CLUTO[7]; it is a kind of K-means method. Repeated Bisection is suitable for short sentences[5]. Topics of tweets differ from diversity and granularity. Actually, it is difficult to ascertain the number of clusters. We use the number of clusters automatically using a threshold: the dividing point of bayon. Furthermore, Repeated Bisection is hard clustering; some garbage clusters exist, which are sparse clusters. We delete such garbage clusters, which are under the threshold of similarity between the center vector of a cluster and the vector of tweets. After deleting garbage clusters, we designate the remaining clusters as "topic clusters" for these analyses.

4.1.2 Creating Topic graph

After dividing the tweets into topic clusters based on the topics, we transform topic clusters into the topic graph using the semantic structure. As described in this paper, we use the category structure of Wikipedia as a semantic structure because the category structure of Wikipedia includes new words of many kinds. Tweets also use new words of many kinds.

(a) Creating the smallest topic graph

First, we transform a topic in a topic cluster into a smallest graph, which consists of a topic in a cluster and its high-level semantics. If a topic cluster has multiple topics, we transform a topic cluster into a multiple topic graph. As described in this paper, we call this topic graph "the smallest topic graphSTG_j." j is a number of a topic. A leaf node of all STG_j is the topic$C = \{c_1, c_2, \ldots, c_n\}$ (show Fig. 4). STG_j is a labeled graph. The label of the leaf node is a topic of the topic cluster. Non-leaf nodes in STG_j are high-level semantics of a leaf nodec_x. We regard the semantics of c_x as a category of Wikipedia, which means that a label of non-leaf nodes in STG_j are semantics of c_x. When we extract high-level semantics of a c_x from Wikipedia, we use the Wikipedia category link information database[2]. We regard category s_{xi} as a high-level semantic of c_x. In addition, i is a number of category of c_x. We also extract high-level semantic s_{xim} of s_{xi} from the Wikipedia category link information database. We create $n-th$ high-level semantics to create the smallest topic graph. As described in this paper, n is 2.

In this time, the category words which contain "Wiki," "stub" and "user" are deleted from the category, because these words are Wikipedia's system words. These are inappropriate for using high level semantics. We also delete "Living people" and "XX-language surnames," which have no important meaning. Then, the leaf node of STG_j is a topic of topic cluster. Non-leaf nodes of STG_j are high-level semantics of a leaf node (topic). For example, there are two topics of "Ichiro" and "Masahiro Tanaka." First, we create the smallest topic graph about "Ichiro" as c_1. We search for articles about "Ichiro" from Wikipedia. We extract categories, which are "American League stolen base champions" as s_{11}, and "Japanese Major Leaguers" as s_{12}, as high-level semantics from the Wikipedia category link information database. We also extract two-hop high-level semantics of topic from the Wikipedia category link information database. Then we create the smallest topic graph related to "Ichiro" (c_1). Next, we search each article about "Masahiro Tanaka" (c_2) from Wikipedia. We also extract categories which are "Japanese Major Leaguers" (s_{21}) and "Olympic baseball players of Japan" (s_{22}) as high-level semantics. Then we extract two-hop high-level semantics of topics. Finally, we create the smallest topic graph related to "Masahiro Tanaka."

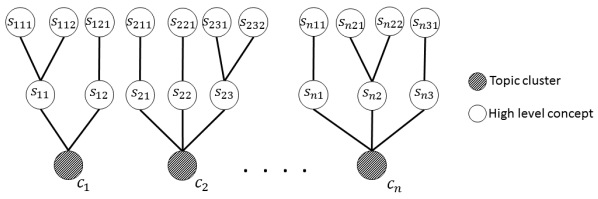

Figure 4: Smallest topic graph

The number of a smallest topic graph is a total number of topics of all topic clusters. After we create all smallest topic graphs, we create topic graphs based on joining at the same nodes.

(b) Creating a topic graph

Next, when the same label exists in STG_j and STG_k, we join STG_j and STG_k to create the topic graph. There are two types of joining: mutual joining of non-leaf-nodes (high-level semantic) (Fig. 5(a)), and joining between a leaf-node (topics) and a non-leaf-node (Fig. 5(b)). For example, in Fig. 5(a), c_1 is "Ichiro" and S_{12} is "Japanese Major Leaguers," c_2 is "Masahiro Tanaka" and S_{21} is also "Japanese Major Leaguers." The labels of S_{12} and S_{21} are the same. Then we join S_{12} and S_{21}. In Fig. 5(b), c_1 is "Japanese Major Leaguers," and c_2 is "Major Leaguers." Also, S_{12} is a high-level semantic of c_1: "Major Leaguers." In this case, the labels of c_2 and S_{12} are the same label. Consequently, we join two nodes.

In this way, we join the nodes of all smallest topic graphs which have the same label to other smallest topic graphs, and create a topic graph. In the topic graph, we designate a node for which the label is a topic as a "topic node" and a node for which the label is only high-level semantic as a "semantic node." A topic node includes tweets for which the topic is the label of the topic node. A semantic node does not include tweets, but only a label of the node (high-level semantic). When we join a topic node and a semantic node in the smallest topic graph, the node becomes a topic node. All leaf nodes of a topic graph are topic nodes. The topic graph presents the topic granularity: a topic that is a greater distance from root node is a more detailed topic than another topic which is a shorter distance from the root node. Consequently, leaf nodes are the most detailed topics in the topic graph.

4.2 Determining Type of Topic based on time interval

We determine topic nodes in a topic graph as known topics or unknown topics. First, we check the time stamp of tweets in the topic nodes. If a time stamp of a tweet is the browsing time of a browsing user, then the tweet becomes a known tweet. However, if a time stamp of tweet is non-browsing time of a browsing user, then the tweet becomes an unknown tweet. After we extract each time stamp of a tweet, we next determine a type of topic node. When at least one known tweet includes a topic node, the topic node becomes

[1]https://code.google.com/p/bayon
[2]http://dumps.wikimedia.org/jawiki

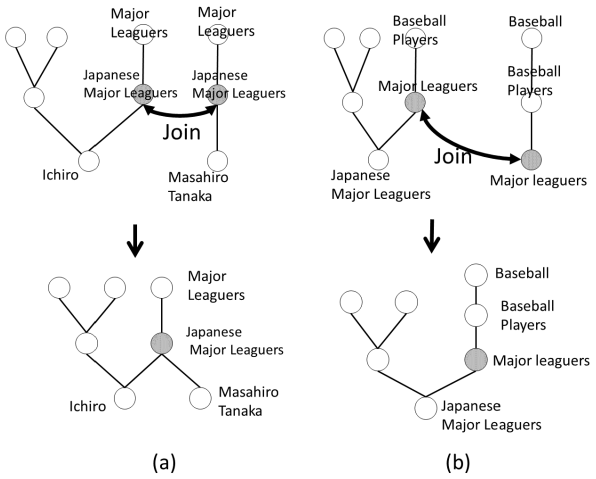

Figure 5: Type of joining in smallest topic graph

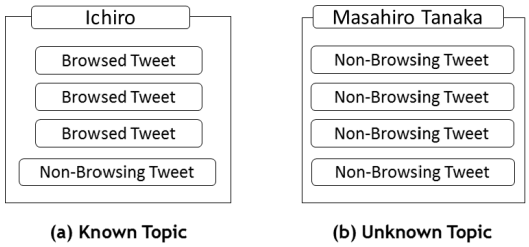

Figure 6: Example of topic types in the topic graph

a known topic. When all tweets in a topic node are unknown tweets, the topic node becomes an unknown topic. Figure 6 presents an example of a type of topic node in the topic graph. In Fig. 6, the topic cluster "Ichiro" includes both browsed tweets and non-browsed tweets. Therefore "Ichiro" is a known topic. In contrast, "Masahiro Tanaka" includes only browsed tweets. Therefore "Masahiro Tanaka" is an unknown topic.

We determine all topic nodes for topics of two types.

4.3 Presenting missing tweets based on granularity and time interval

After creating the topic graph and determine the type of topic node, we ascertain what contents are present for a browsing user based on the type of topic graph.

Our content-presenting semantic is that it is good for the browsing user to present an outline of the topics in the case of a known topic. Then we present a structure of a topic graph of the missing information to the user. For an unknown topic, it is difficult for a browsing user to understand all contents of a topic clearly because the tweets were tweeted during a time when the browsing user is unable to browse. We present details of a topic graph. The following and Fig. 7 present the type of topic graph. In Fig. 7, black nodes, both circles and triangles, are topic nodes that include tweets for which the topic is the label of the node. Nodes of black circles are known topics. Nodes of black triangles are unknown topics. Nodes of white circles are semantic nodes. They do not include tweets. Gray circles are browsing areas that include only topic nodes. Gray rectangle areas are also browsing areas, but they consist of topic nodes and semantic nodes.

- Un-joined smallest topic graph
  This case of a smallest topic graph is not joined with other smallest topic graphs. In this case, topics (nodes) are mutually isolated from other topics.
  (a) Known topic (Fig. 7(a))
  The browsing user knows a part of tweets in the topic node, which means the user knows an overview of the structure of the smallest topic graph. We present only topics of the leaf node.
  (b) Unknown topic (Fig. 7(b))
  The browsing user does not know all tweets in the topic node. Therefore, the browsing user does not know the structure of the smallest topic graph. We present details of the topics. Then we present all topic structures of the topic graph and all tweets in the topic node.

- Joined smallest topic graph
  In this case, there are two cases which are joined between semantic nodes and joined between topic node and semantic node (see Fig. 5).
  – Between semantic node
  In this case, the topic nodes differ, but they have the same semantic nodes.
  (c) All known topics (Fig. 7(c))
  This case is that for which all topic nodes are known topics. This case is almost identical to that of type (a), but we regard the common high-level semantic as important. We present a label of all topic nodes and the label of common semantic nodes.
  (d) All unknown topics (Fig. 7(d))
  This case is that for which all topic nodes are unknown topics. The browsing user does not know the content of the topic graph at all. This case is the same as type (B). We present all topic structures of the topic graph and all tweets in all topic nodes.
  (e) Known topic and unknown topic (Fig. 7(e))
  Basic concepts of this case are mixed types (a) and (b). The browsing user knows part of the structure of the topic graph. We present labels of known topics along with tweets in unknown topics and labels of semantic nodes which are parent nodes of unknown topics.
  – Between topic node and semantic node
  (f) Known topic & Known topic (Fig. 7(f))
  In this case, the parent node of known topic is also a known topic. The browsing user knows the topic nodes. The user knows an overview of the structure of topic graph. We present labels of topic nodes and parent topic nodes.
  (g) Known topic & Unknown topic (Fig. 7(g))
  In this case, the parent node of a known topic is an unknown topic. The browsing user knows the topic node, but the user does not know

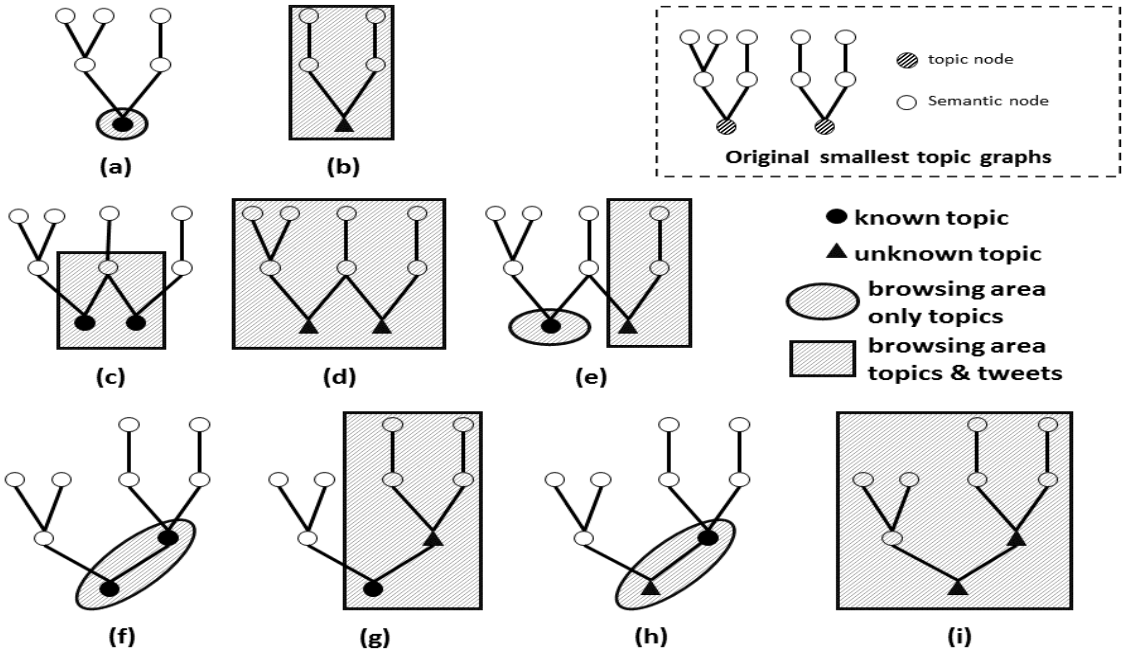

Figure 7: Topic graph types

the parent of the topic node. We consider that the browsing user does not know the topic graph structure well. Therefore, we present a label of the topic node along with labels and tweets of parent topic nodes.

(h) Unknown topic & Known topic (Fig. 7(h))
In this case, the known topic is the parent node of the unknown topic. We regard the browsing user as knowing almost the entire structure of the topic graph. Then we present the tweets and label of the topic node, which is an unknown topic, and the label of the parent topic node, which is a known topic.

(i) Unknown topic & Unknown topic (Fig. 7(i))
In this case, we expect that the browsing user does not know the overall structure of the topic graph. We present all tweets of the unknown topics and all labels of all nodes of the topic graph.

5. EXPERIMENT

We conducted experiments of two types to obtain the benefits of our proposed method.

5.1 Experiment 1: Clustering Method

We conducted an experiment to obtain whether repeated bisection is suitable for clustering of the tweets of followees.

5.1.1 Condition

Conditions of experiment 1 are the following:

- Parameter:
  divided point of bayon is 1.0
  threshold of cosine similarity is 0.5.
- Data sets
  - Number of users: 5
  - Number of tweets: 1000 tweets/user total is 5000 tweets.
  - Feature of users:
    User A: interest topics: “anime,” “game”
    He tweets many short tweets and nonsensical tweets.
    User B: interest topics: “anime,” “idol”
    He tweets specific idol names and idol groups.
    User C: interest topics: “soccer”
    He tweets long tweets related to soccer games and players.
    User D: interest topics: “soccer,” “baseball,” “game”
    He always tweets daily things.
    User E: interest topics: “music,” “radio”
    Many of his tweets include local names and building names.
  - Correct answer:
    We extract correct answers manually.

5.1.2 Results and Discussion

Table 1 shows results of clustering. We regard the correct data is by hands and we calculate precision. The precision is greater than 0.7 in all users in the Table 1,the results are good. Especially, results of User B, User D and User E are high precision.The reason is that these users tweeted with a lot of proper nouns such as human name, place name, and music title, then there are many feature words which are used for clustering in their tweet. It is easy to cluster these tweets by using the feature words. However, results of User A and User C are not high precision. The reason is that their tweets are too short and there are many slang and

Table 1: Results of clustering

	Number of tweets	Number of cluster	Number of tweets in the largest cluster	Precision
User A	325	97	14	0.708
User B	353	47	23	0.912
User C	583	171	7	0.762
User D	487	124	14	0.891
User E	469	121	12	0.938
Average	443.4	112	14	0.842

Table 2: Example of Topics in each Cluster

User name	Cluster ID	Topic 1	Feature value	Topic 2	Feature value
User A	61	Speedcore	0.999	t+pazolite	0.023
	36	familymart	0.978	chills caught in sleep	0.211
	65	comicmarket	0.963	Take it easy	0.103
	53	?? (doubke question marks)	0.962	Tear	0.157
	41	Ramen	0.962	Ramen Jiro	0.160
User B	51	ISUCA	0.986	Kohime	0.121
	39	Keyhole Turtleneck	0.986	Elithabeth	0.067
	27	anisama	0.985	Audience Camera	0.087
	13	Otome	0.969	ISUCA	0.201
	74	ISUCA	0.959	ISUCA tremendous	0.186
User C	10	excitement	0.876	eroticism	0.340
	14	bundesliga	0.856	role	0.333
	17	Osaka	0.839	somebody else problem	0.164
	19	gekisaka	0.836	Zenit	0.176
	21	concern	0.831	new world	0.287
User D	32	Tigers	0.974	Central League	0.175
	15	Dale	0.971	TsumTsum	0.239
	93	Koshien	0.971	Shimabukuro	0.0962
	151	Gomez	0.970	RBI	0.136
	113	Part time job	0.969	Breaktime	0.068
User E	148	MINAMI WHEEL	0.980	Carbonara	0.099
	123	Nigata	0.962	Doshite	0.084
	129	Fukuoka	0.962	fmfukuokafm	0.136
	93	Nagoya	0.959	Beautiful Sky	0.113
	14	Tweet	0.958	I Love You	0.285

abbreviation words in their tweets, it is difficult to extract feature words to clustering.

Table 2 presents results of clustering of the top five clusters based on respective users. In this table, topics of user C are not the same as his interest topics; the topics are almost entirely general words such as "excitement" and "concern." The reason is the problem of dictionary of Japanese language morphological analysis. His tweets are not good for Japanese language morphological analysis. We cannot extract feature words. Then the tweets are clustered by general words. From these results, our method of extracting feature words and clustering methods is good for the clustering the followees' tweets.

5.2 Experiment 2: Creating topic graphs

We conducted an experiment to ascertain whether creating the smallest topic graphs is beneficial or not.

5.2.1 Conditions

The conditions of experiment 2 are the following:

- data: 2335 tweets that contain obvious topics clustering by experiment1s method, the number of cluster is 152 topic clusters.
- The number of topic graph : 299.
- Subjects: Nine men and women in their 20s.

The flow of experiment is

1. We extract the top two topics from each cluster. Then we extract categories of topics from Wikipedia.
2. We create the smallest topic graph of each clusters.
3. We join the same node of smallest topic graphs and create topic graphs. In this experiment, we joined only two clusters in each topic graph.
4. Subjects show tweets in the topic graph, and judge the joined node is good or not as three choices (0, bad; 1, middle; 2, good).

5.2.2 Results and Discussion

Table 3 shows the experiment results: we select three(J-league, Anime, and Macroeconomics) are good results; the other three(Help, Society, and Registration of mark) are bad. Two points of difference exist between good results and bad results. First, if the cluster topic is specific, the joining is successful. For example, each cluster of J-league club(which is Japanese professional football league name), Anime, Macroeconomics have specific topics. However, the clusters of help, society, registration of marks have wide topics. For them, clustering was not successful. Second, high-level semantic in the bad case have a large number of subordinate semantics, which is over 100. The topic graph is not successful if the high-level semantic has wide semantics. In the near future, we must consider such cases.

Table 3: Example of results of experiment2

High-level semantic	Number of subordinate semantic	Topic in cluster A	Topic of cluster B	Average of value
J league club	54	Bellmare,Shonan BMW stadium	Urawa Reds, urawareds	1.89
Anime	95	TV anime, NeversayNever(Music title)	Seiyu-Paradise(Music title), NeversayNever(Music title)	1.78
Macroeconomics	79	primary balance, second report	macro economy, report of municipal government	1.56
Help	107	football player, baseball player	Mamers(product name), schedule	0.00
Society	149	tennis, condition	energy, Yamaga-member(team name)	0.00
Registration of mark	414	Softbank, rute 1	Cancam(Magazine name), energy policy	0.00

6. CONCLUSION AND FUTURE WORK

As described in this paper, we propose a method to extract missing information that a user missed: it is important and unknown information from the user's timeline on Twitter. The method also browses a summary of these numerous tweets automatically. First, we cluster the followee's tweets to find the user's missing information. Then we extract their topics. Next we create a topic graph based on the semantic structure from Wikipedia. We determine the topic type from followee's topic graph based on the browsing time interval. Then we extract missing information and present the information to the users. We conducted experiments of two types to measure the feasibility of clustering for followee's tweet and to assess the availability of topic graphs. These results demonstrate that our method of extracting topics and clustering methods is good for clustering of the followees' tweets. Results show that our topic graph is almost feasible. However, for cases in which the topics are not specific, our topic graph is unsuitable. In the near future, we expect to conduct experiments to improve content presentation based on topic granularity. We must also develop a user interface for our systems and conduct user experiments.

In future work, first, we expect to improve the method of extracting future words and high-level semantics based on results of the experiments described herein. Next, we plan to create a user interface and experiment to demonstrate that our method and user interface are appropriate for browsing missing information.

7. ACKNOWLEDGMENTS

This work was partially supported by Research Institute of Konan University, and by JSPS KAKENHI Grant Number 26330347.

8. REFERENCES

[1] H. Becker, M. Naaman, and L. Gravano. Beyond trending topics: Real-world event identification on twitter. *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, pages 2695–2697, 2011.

[2] M. S. Bernstein, B. Suh, L. Hong, J. Chen, S. Kairam, and E. H. Chi. Eddi: Interactive topic-based browsing of social status streams. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 303–312, New York, NY, USA, 2010. ACM.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

[4] M. Cataldi, L. Di Caro, and C. Schifanella. Emerging topic detection on twitter based on temporal and social terms evaluation. pages 4:1–4:10, 2010.

[5] S. Hanai and A. Nadamoto. Clustering for similar recipes by using cooking ingredient. *IEICE technical report*, 114(204):47–52, 2014.

[6] L. Hong and B. D. Davison. Empirical study of topic modeling in twitter. *Proceedings of the First Workshop on Social Media Analytics (SOMA).*, pages 80–88, July 2010.

[7] G. Karypis. Cluto - a clustering toolkit. *Dept. of Computer Science*, May 2002.

[8] S. P. Kasiviswanathan, P. Melville, A. Banerjee, and V. Sindhwani. Emerging topic detection using dictionary learning. pages 745–754, 2011.

[9] K. Lee, D. Palsetia, R. Narayanan, M. M. A. Patwary, A. Agrawal, and A. Choudhary. Twitter trending topic classification. *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*, pages 251–258, December 2011.

[10] J. MacQueen. Some methods for classification and analysis of multivariate observations. *5th Berkeley Synap. Math. Statist*, 1:281–297, 1967.

[11] W. Magdy, A. Ali, and K. Darwish. A summarization tool for time-sensitive social media. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 2695–2697, New York, NY, USA, 2012. ACM.

[12] M. Mathioudakis and N. Koudas. Twittermonitor: Trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, SIGMOD '10, pages 1155–1158, New York, NY, USA, 2010. ACM.

[13] M. Michelson and S. A. Macskassy. Discovering users' topics of interest on twitter: A first look. In *Proceedings of the Fourth Workshop on Analytics for Noisy Unstructured Text Data*, AND '10, pages 73–80, New York, NY, USA, 2010. ACM.

[14] Z. Ren, S. Liang, E. Meij, and M. de Rijke. Personalized time-aware tweets summarization. *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval.*, pages 513–522, 2013.

[15] K. D. Rosa, R. Shah, B. Lin, A. Gershman, and R. Frederking. Topical clustering of tweets. *The 3rd Workshop on Social Web Search and Mining (SWSM).*, 2011.

[16] K. Sasaki, T. Yoshikawa, and T. Furuhashi. Online topic model for twitter considering dynamics of user interests and topic trends. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).*, pages 1977–1985, October 2014.

[17] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu,

and M. Demirbas. Short text classification in twitter to improve information filtering. pages 841–842, 2010.

[18] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques. *In 6th ACM SIGKDD,World Text Mining Conference*, 2000.

[19] T. Xu and D. W. Oard. Wikipedia-based topic clustering for microblogs. *Proceedings of the American Society for Information Science and Technology*, pages 1–10, 2011.

[20] W. X. Zhao, J. Jiang, J. H. J. Weng, E. Lim, H. Yan, and X. Li. Comparing twitter and traditional media usingtopic models. *In Proceedings of the 33rd European Conference on Information Retrieval.*, 2011.